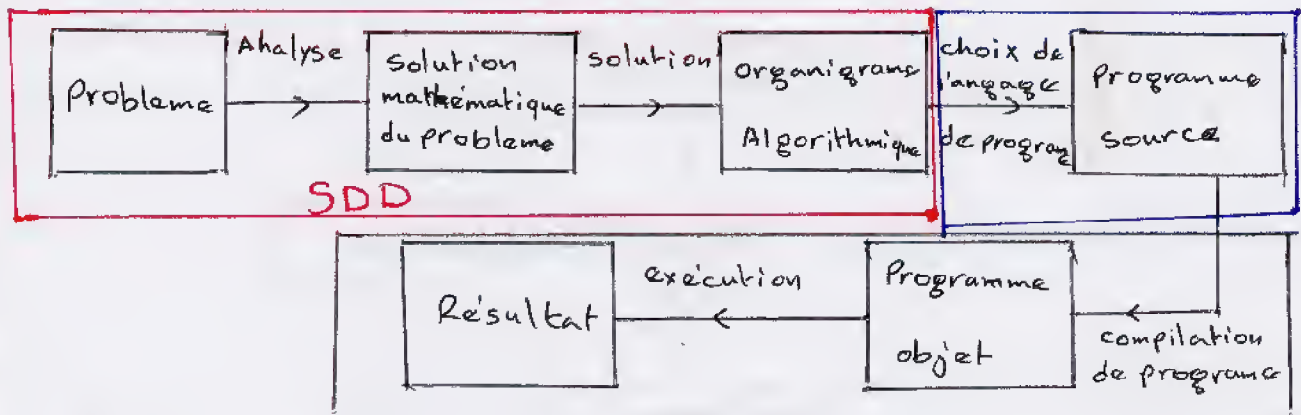


Déroulement d'un programme

Analysé un problème revient

- ① À déterminé les paramètre d'entrée
- ② À déterminé les paramètre de sortie
- ③ À déterminé la méthode de solution (Le traitement)

Exemple: Délébère pour un étudiant

- Ⓐ paramètre d'entrée: N° étudiant, Nom et prénom étudiant, Notes étudiant
- Ⓑ paramètre de sortie: N° étudiant, Nom et prénom étudiant, Notes étudiant, moyen général étudiant, Résultat, La mention, Le Rang (classement)
- Ⓒ Le traitement: Calculé la moyenne générale
 - Résultat: Résultat admis ≥ 10 Résultat ajourné < 10
 - Déterminé La mention: * Passable (10-12) * Bien (12-14)
 - * Très bien (14-16) * excellent (16-18)

Pour proposer une solution il existe 2 forme:

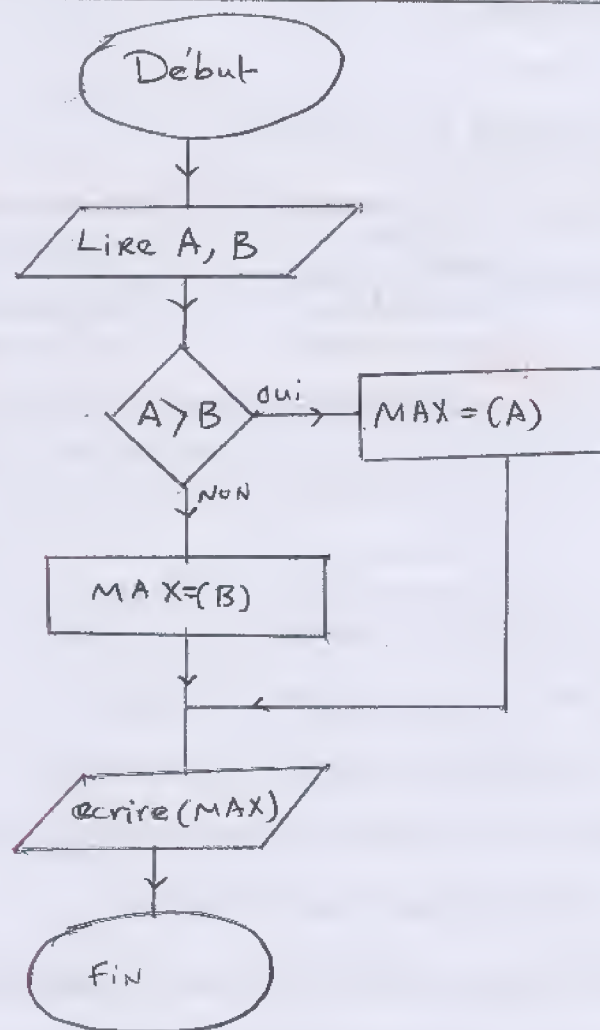
Forme I: Organigramme: Cette forme est très simple à comprendre

C'est une méthode de solution sous un forme d'un schéma

Exemple:

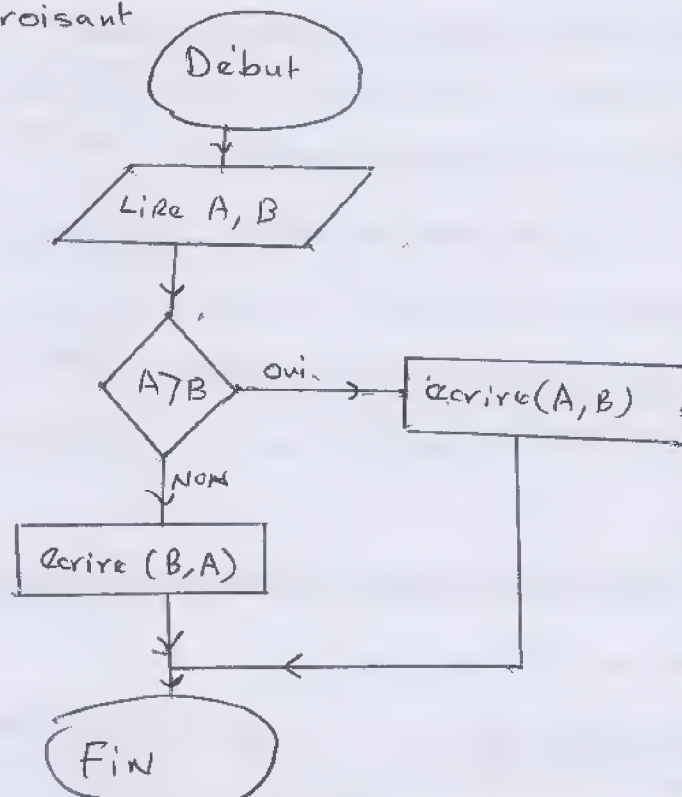
Donnée l'organigramme qui permet de calculé le maximum (MAX) de entier Le "A" et "B"

- Ⓐ entrée: A, B
- Ⓑ sortie: Maximum (MAX)



Exercice N°2

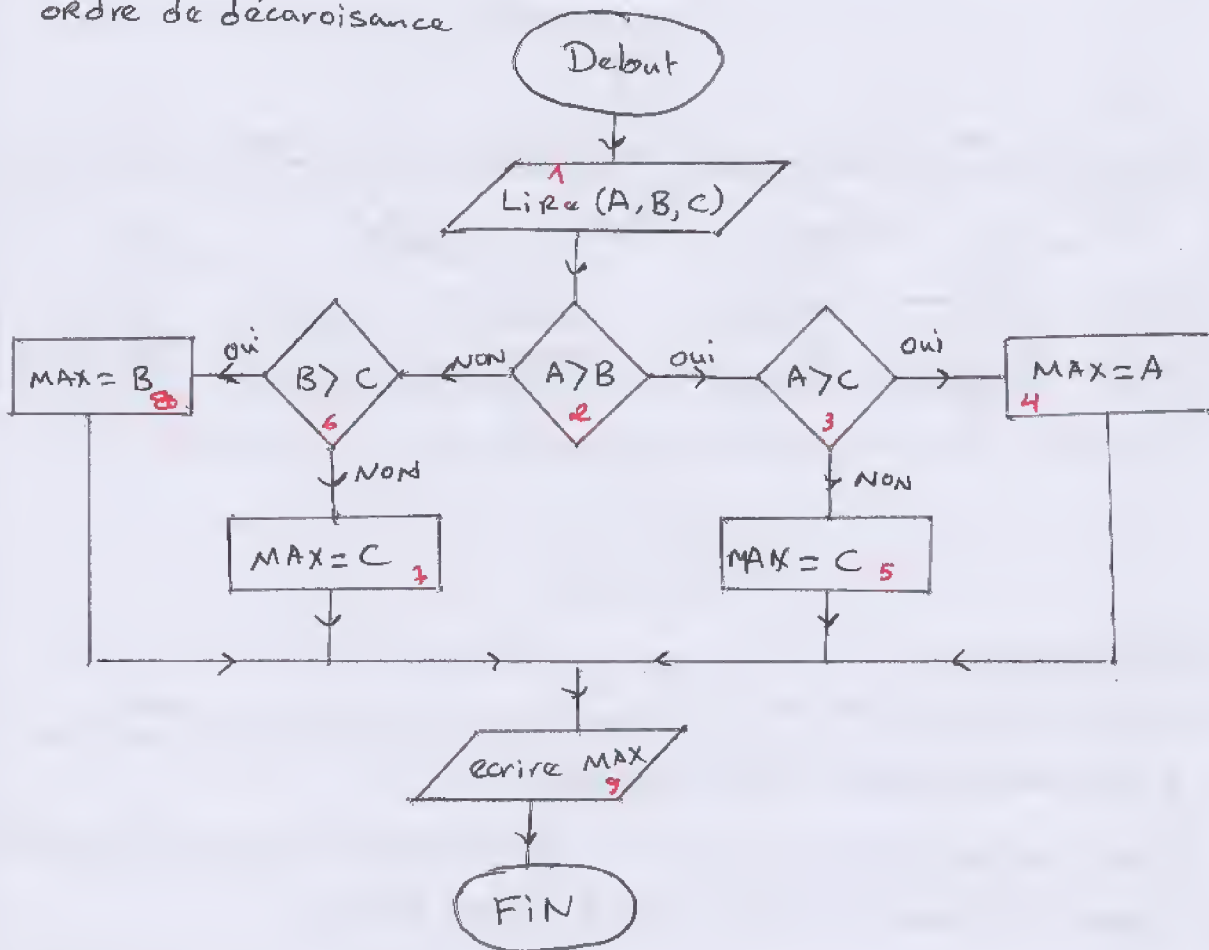
① Donner l'organigramme qui permet d'afficher les deux nombres A et B par ordre de décroissant



Proposition SDD

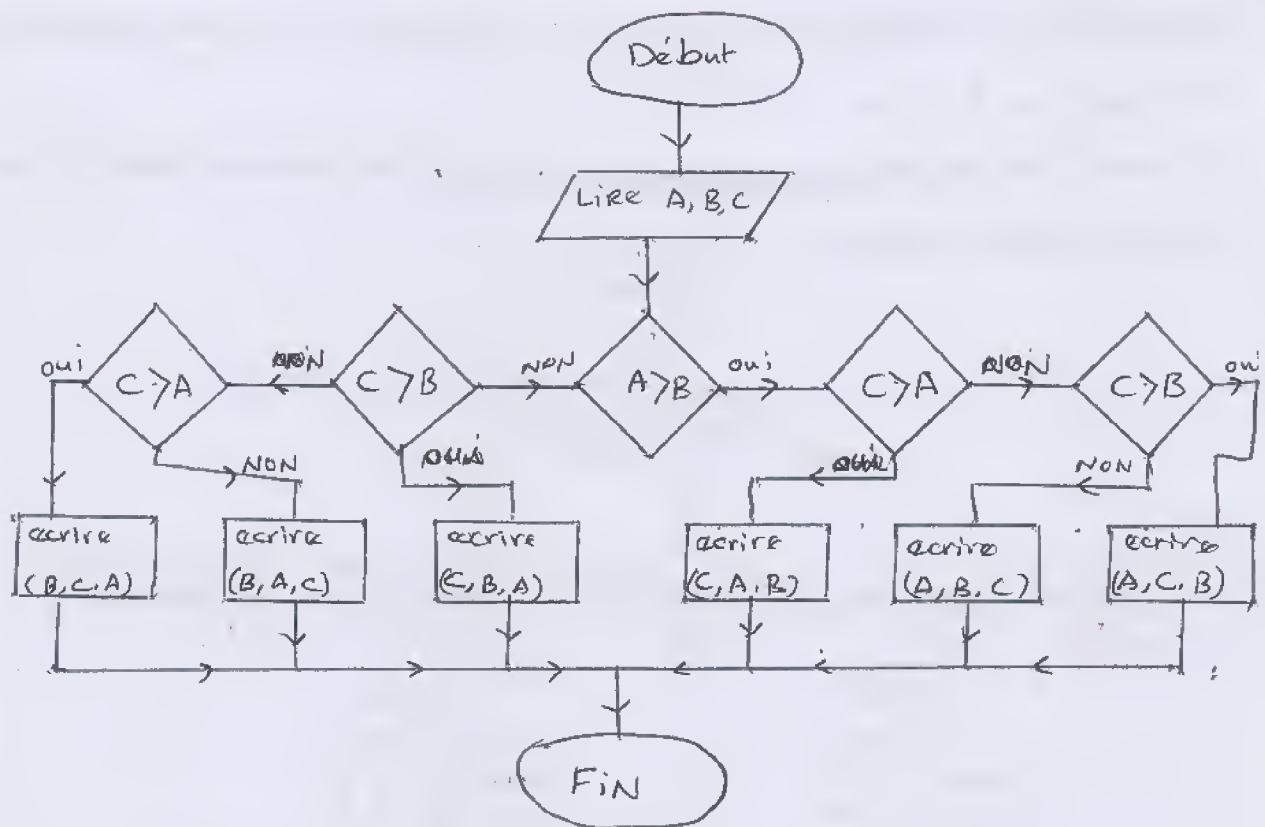
② Donner une procédure qui permet de calculer le maximum (MAX) entre trois nombre A, B et C

③ Donner l'organigramme qui permet d'afficher trois nombre (A, B et C) par ordre de décroissance



Remarque:

- ⊕ Ce programme de 9 instruction et exécute 5 instruction
- ① Ce programme est de 9 instruction,
- ② Ce programme ~~est de~~ exécute que 5 instruction.
- ③ Les instruction 1, 2 ~~et~~ 9 sont toujours exécuté.

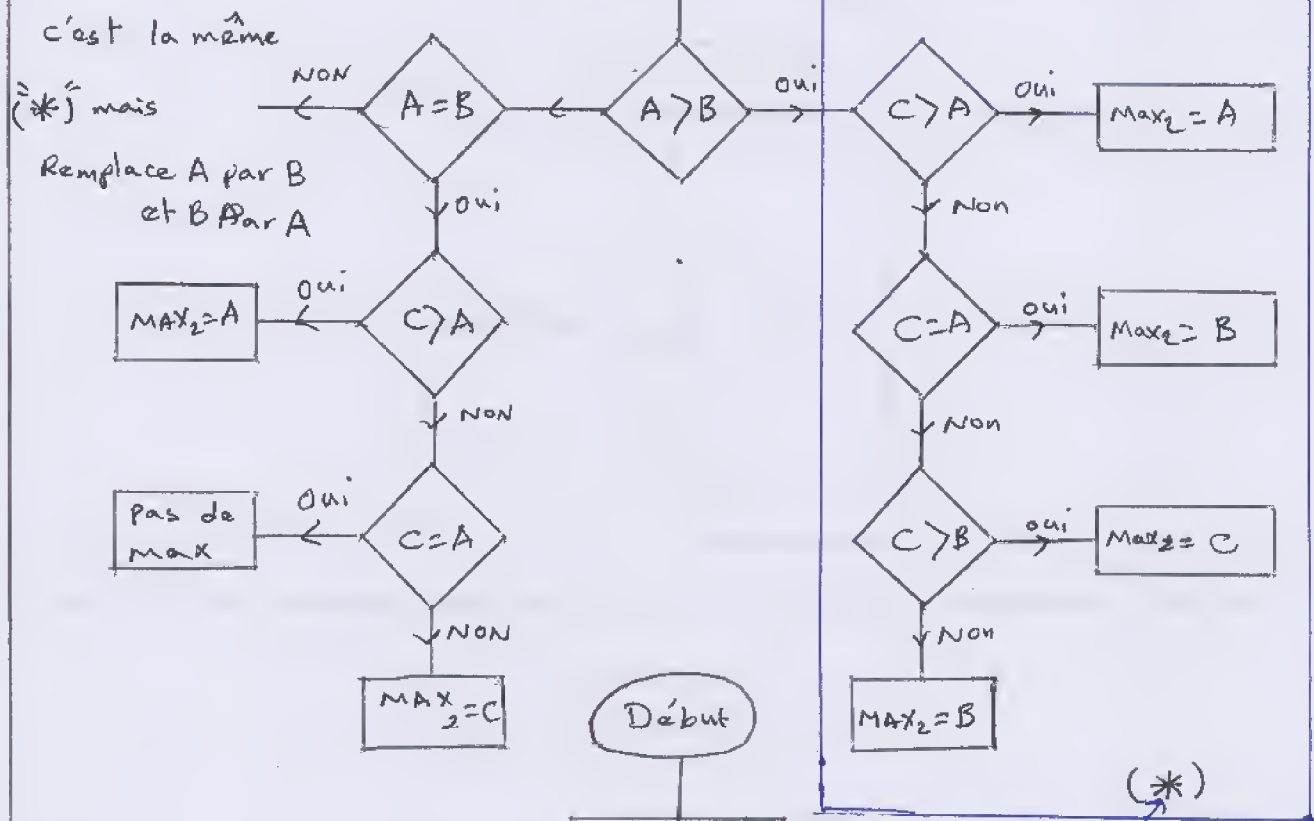


Exercice N° 3

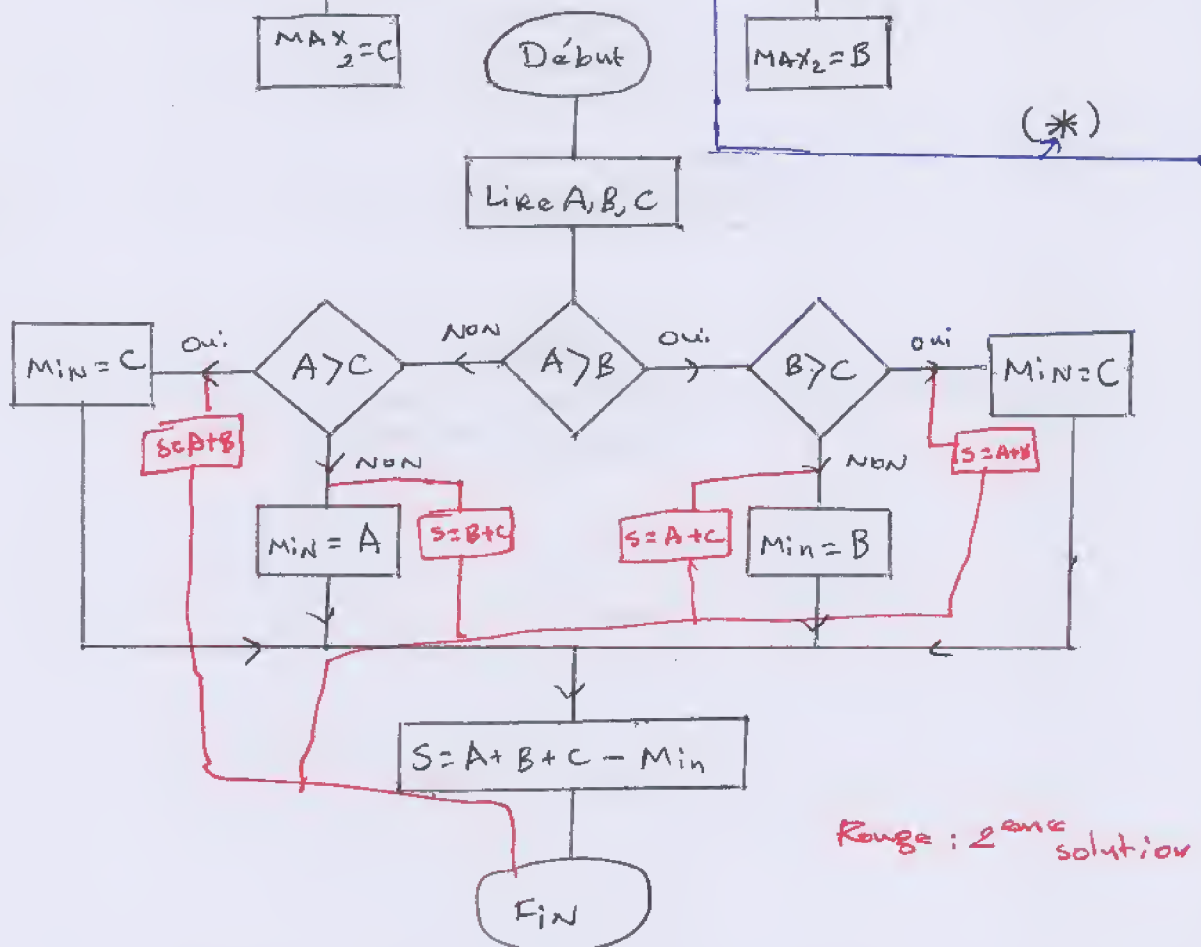
- ① Donner l'organigramme qui permet de calculer le 2^{ème} maximum de 3 Nombre A, B, C ($\text{Max}_2 < \text{Max}_1$)
- ② Donner l'organigramme qui permet de calculer la somme des nombre des deux plus grand Nombre des 3 Nbre A, B, C
- ③ Donner l'organigramme qui valide une horloge dans deux cas suivant 24H et 12H
- ④ Donner l'organigramme qui permet d'une heure quelconque correcte h.m.s de donner l'heure suivante (+ 18 seconde) dans les deux cas

Solution

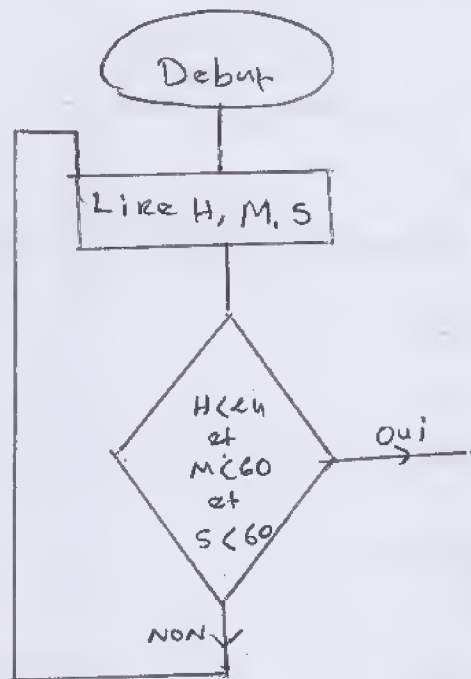
①



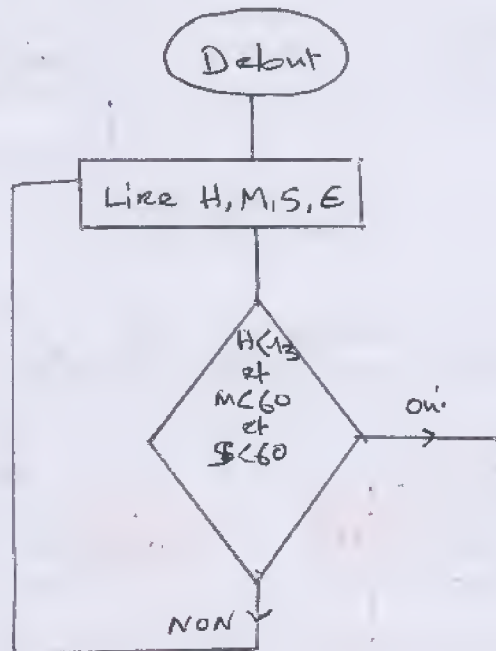
②



③ * Une procédure qui permet de valider une horloge de 24 heures

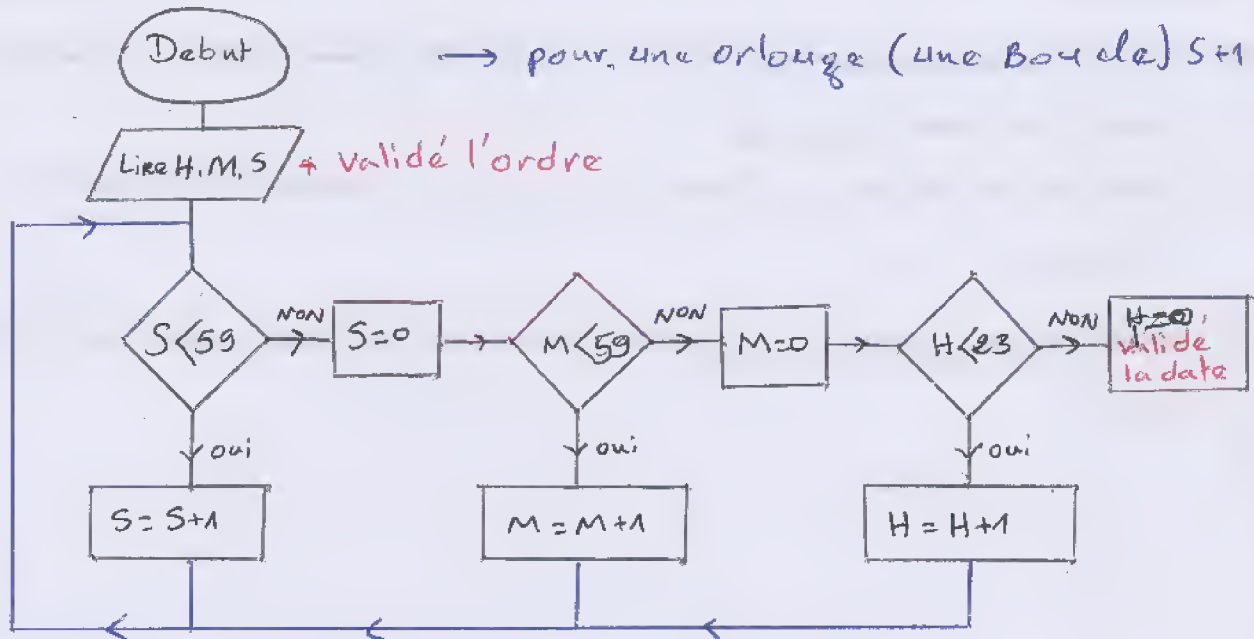


* Une procédure qui permet de valider une horloge de 12 heures

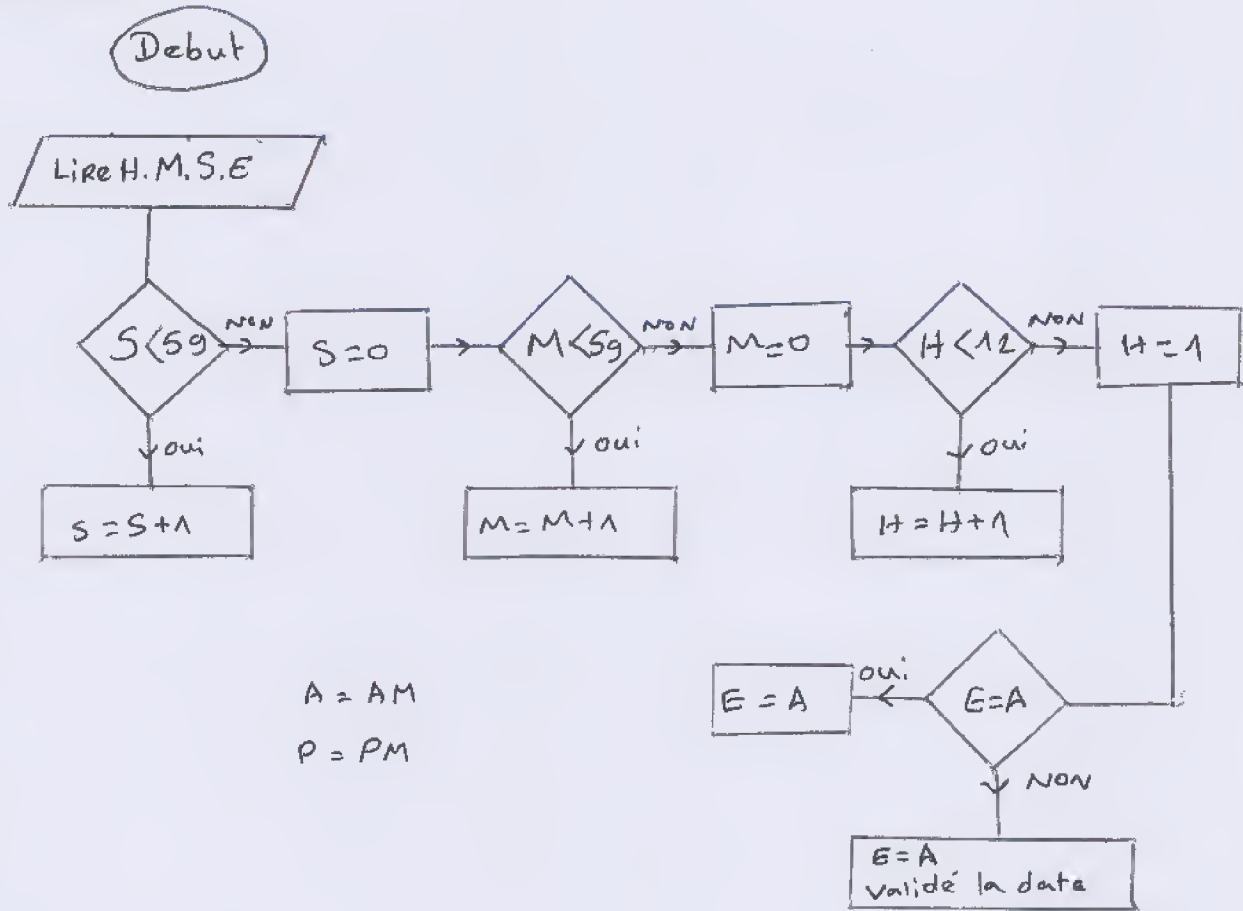


④ * l'organigramme qui permet d'une heure quelconque correcte h. m. s de donner l'heure suivante (+ 1 seconde) de 24 h et 12h

* Cas de 24 h



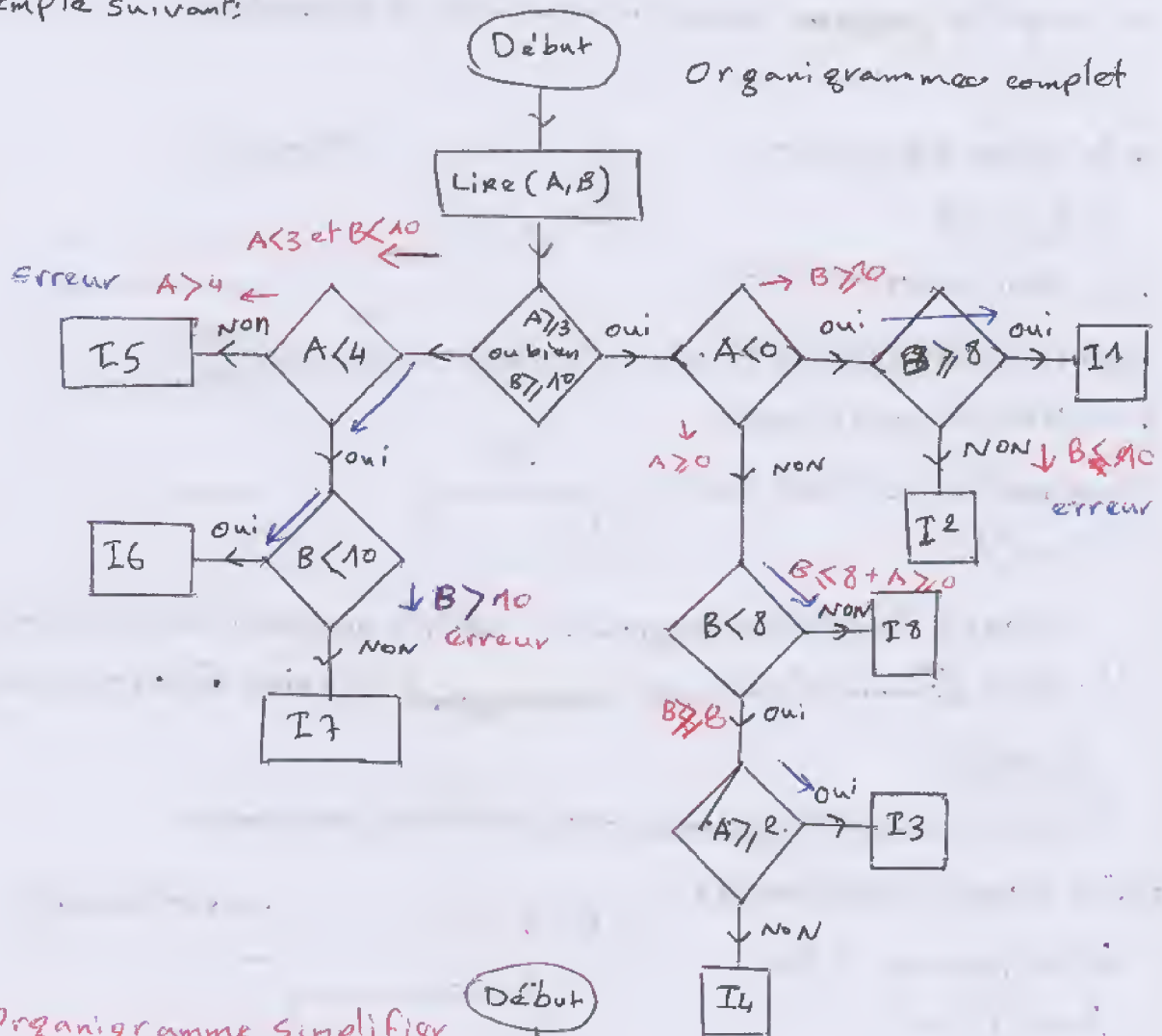
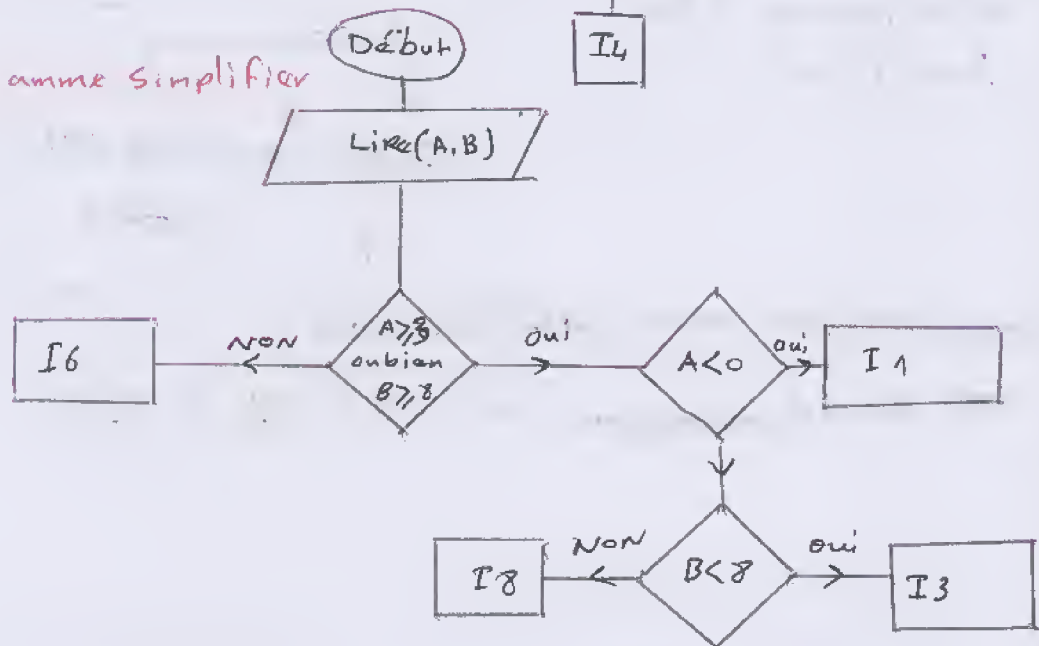
* Cas de 12h



- ⑤ Donner une procédure qui permet de valider une date quelconque
- ⑥ Donner une procédure d'associer une date quelconque la journée correspondante
- ⑦ Donner la procédure qui permet à partir d'une date quelconque la date de ~~l'année~~ demain
- ⑧ Donner la procédure qui permet de calculer le compte à rebours à -1 seconde
- ⑨ Donner la procédure qui permet de calculer la durée entre les 2 dates quelconques
- ⑩

Simplification des organigramme

C'est l'élimination des instructions qui ne saire à rien comme exemple suivant:

Organigramme simplifier

1. The first part of the paper is devoted to a general discussion of the problem of the existence of solutions of the system of equations

$$\begin{cases} \Delta u = f(x, y, u, v) \\ \Delta v = g(x, y, u, v) \end{cases} \quad (1)$$

where f and g are continuous functions of x, y, u, v in the domain D of the plane. The boundary conditions are assumed to be of the form

$$u = \phi(x, y), \quad v = \psi(x, y) \quad \text{on } \partial D$$

where ϕ and ψ are continuous functions on the boundary ∂D . The problem is to find a solution of (1) which satisfies the boundary conditions (2).

The second part of the paper is devoted to a study of the problem of the uniqueness of solutions of the system (1) under the boundary conditions (2).

The third part of the paper is devoted to a study of the problem of the existence of solutions of the system (1) under the boundary conditions (2).

SDD

* Dans ce cas la procédure est convergente si le ~~I~~ $I \leq 50$ est paire (avec Le Noir)

→ Dans Le cas de (Rouge)

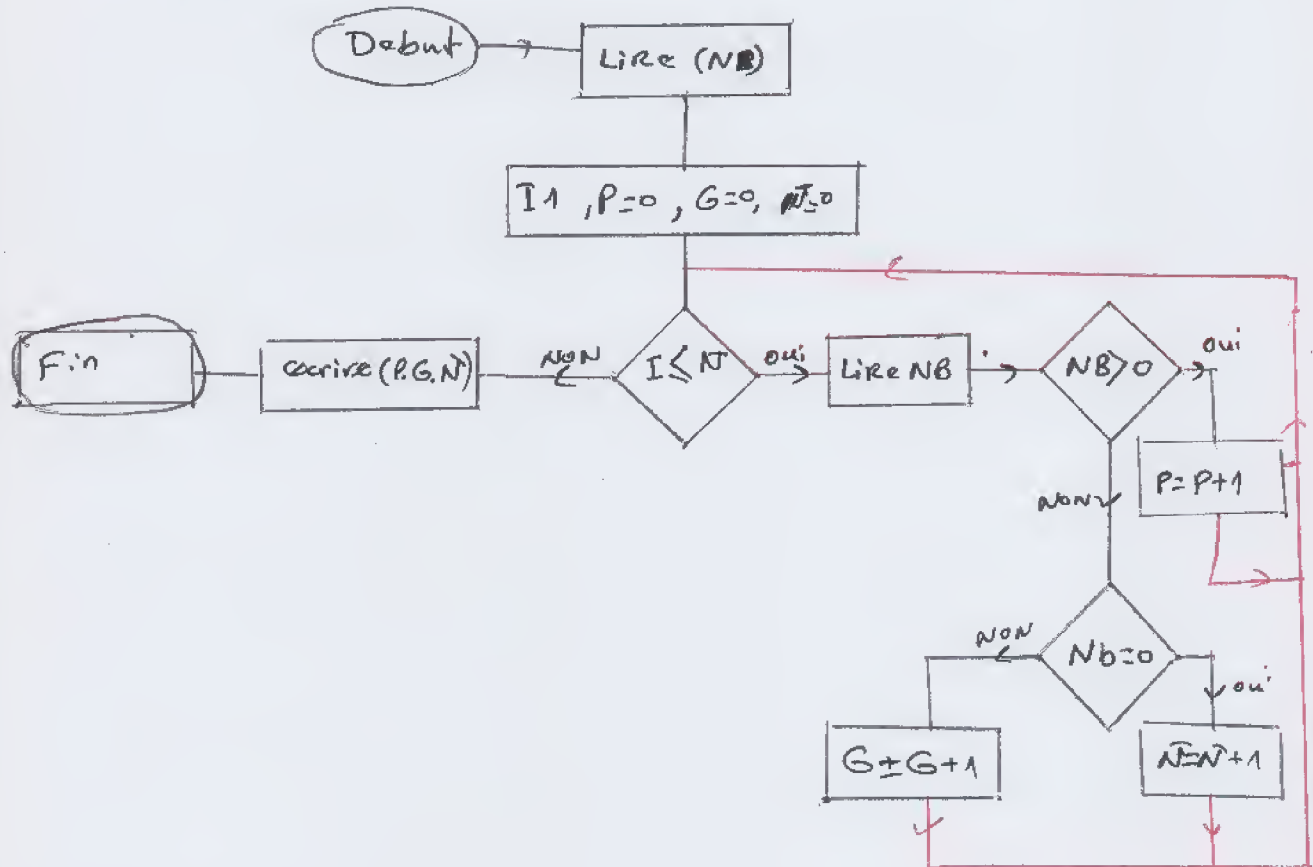
La procédure est convergente

si $2A > A \Rightarrow 2A - A > 0 \Rightarrow A > 0$

$A > 0$

Exercice :

Donné une procédure qui permet de lire N Nbre et de compter Le nombre d'élément positive, négative et nule ~~A = A2~~



- Calculé Le maximum
- Calculé Le minimum
- Calculé Le Maximum et le minimum

Les Algorithmes

Définition:

Un Algorithme est composé de trois parties:

① L'entête de l'Algorithme:

Algorithme NOM : Le nom du programme reflète le travail du programme

② La partie déclaration:

Déclarer toutes les variables utilisées dans l'algorithme, comme exemple :

Var A, B = entier;

C = Réel.

Et les types de toutes les variables dans l'algorithme qui sont :

- Constante: C'est une variable qui prend la même valeur dans tout le programme
 - Entier: une valeur entier on lui réserve un seul champ
 - Réel: une valeur réel on lui réserve deux champs (un pour la partie entier et une autre champ pour la partie décimale)
- * On peut affecter un entier à un réel (on lui rajoute juste le 0, → 0) mais un réel ne peut jamais être affecté à un entier.

Les tables d'opérations des variables:

+ - *	E	R
E	E	R
R	R	R

/	E	R
E	R	R
R	R	R

Les erreurs qu'on peut avoir après la compilation selon leur niveau de difficulté de les détacher

① Erreurs lexiques: erreurs d'orthographe

② Erreurs syntaxique: vérifier les parenthèses, les virgules, séparation

(l'expression doit être juste) Le type (au lieu d'entier, faire réel par exemple)

③ Erreurs sémantique: Ce sont des erreurs difficiles à détecter (les valeurs)

• Booléen: Une variable est déclarée booléen (vrai/faux)

* Une variable booléen ne peut prendre que les valeurs (vrais/faux)

TR = Expression

Si TR = Booléen

La valeur de l'expression est booléen

• Tableau: Les vecteurs (sont des tableaux à 1 dimension)

Les matrices ce sont des tableaux à 2 ou plus de dimension

• Caractères: chaîne de caractère, Liste

(e) Ensemble d'instruction (action) entre les 2 mots réservés debut et fin

Debut

| Actions (Instructions)

Fin

Les instructions:

• L'instruction lecture:

Lire (A): Faire rentrer la valeur "A" (on respectant le type de la valeur déclarer)

• L'écriture:

Ecrire (A): permet d'afficher la valeur de A

Ecrire('A =', A, A + B)

l'affichage: A = 3, 11 (Si A = 3, B = 8)

• L'affectation:

La Syntaxe de l'instruction affectation est:

Id = Exp (Identificateur = expression)

exemple:

Dérouler Les procédures suivantes:

Algorithme ESSA1

Var A, B, C : entier

D, E : Réel

F : Booléen

Début

A = 0, 2, -3

Lire (A)

C = 2

B = A + C

C = A - B

D = A - B

E = A/B

F = E > D

① 1^{er} cas: A = 0

0

A

2

B

2

C

-2 0

D

0 0

E

✓

F

② 2^{ème} cas: A = 2

2

A

B

C

D

E

Exercice:

Déterminer les erreurs de l'algorithme suivants en précisant le type de l'erreur:

Algorithme ESSA1

Var A, B, C = entier

D, E = Réel

F = Bool

"Début"

Lire (A)

~~Lire (A+B)~~ erreur syntaxique

C = 2 - A ✓

D = 2 * A ✓

A = D ✗ erreur syntaxique

B = A / C

~~A + B = C~~

~~E = 3x + 4~~

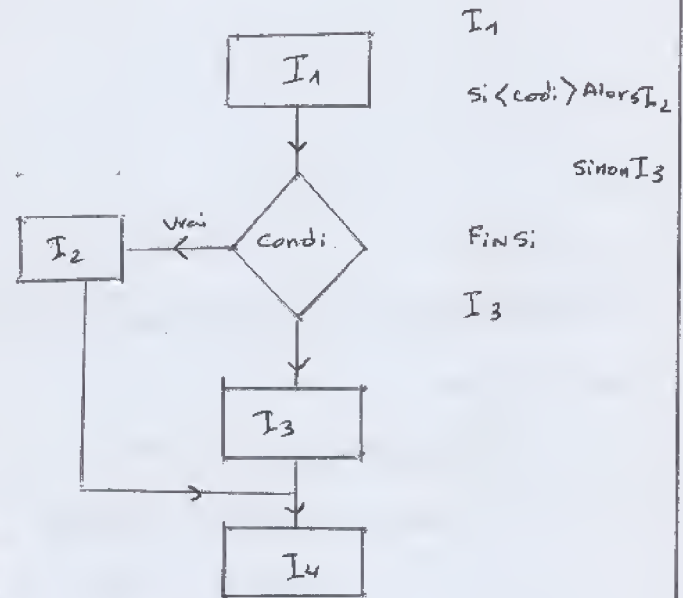
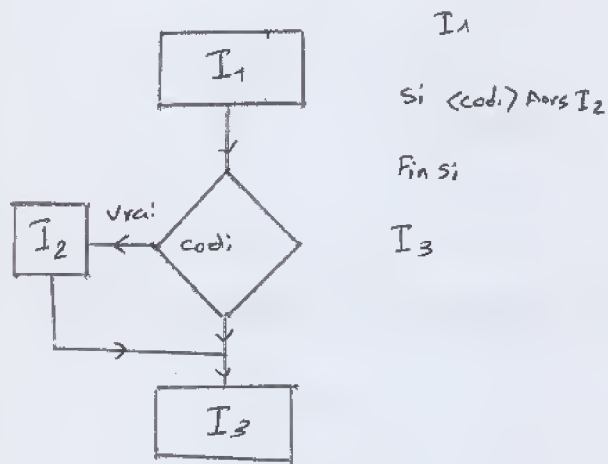
A F = (3 > 4) ou 5

~~F = E + F~~

~~K = 2 * K~~

Fin

* Donner un algorithme qui permet de calculer la moyenne générale d'un étudiant qui prépare 03 modules de coefficient 2, 3, 1



Exercice

exécuter le programme suivant

+ Debut

Lire (A, B)

 $C = A + B$ Si $C > 8$ Alors $A = 2 \times A$

Fin Si

Si $A > 2$ Alors $B = B - A$ Sinon $B = 2 \times B$

Fin Si

Si $A > B$ Alors Si $B > 0$ Alors $B = 2 \times B$

Fin Si

Fin Si

FIN

Exercices:

I. Donner une procédure qui permet de trier des trois nombres A, B, C

* Algorithmes Trier

Var: A, B, C : réel;

Début

écrire ('A = '); Lire(A);

écrire ('B = '); Lire(B);

écrire ('C = '); Lire(C);

Si $A > B$ Alors Si $B > C$ Alors écrire (A, B, C)

sinon Si $A > C$ Alors écrire (A, C, B)
sinon écrire (A, B, C)

Fin Si

Fin Si

Sinon Si $A > C$ Alors écrire (B, A, C)

sinon Si $B > C$ Alors écrire (B, C, A)
sinon écrire (C, B, A)

Fin Si

Fin Si

Fin Si

Fin

II. Donner un algorithme qui permet de résoudre l'équation 1^{er} degré $AX+B=0$

III. Donner un algorithme qui permet de résoudre l'équation de 2nd degré

$$AX^2 + BX + C = 0$$

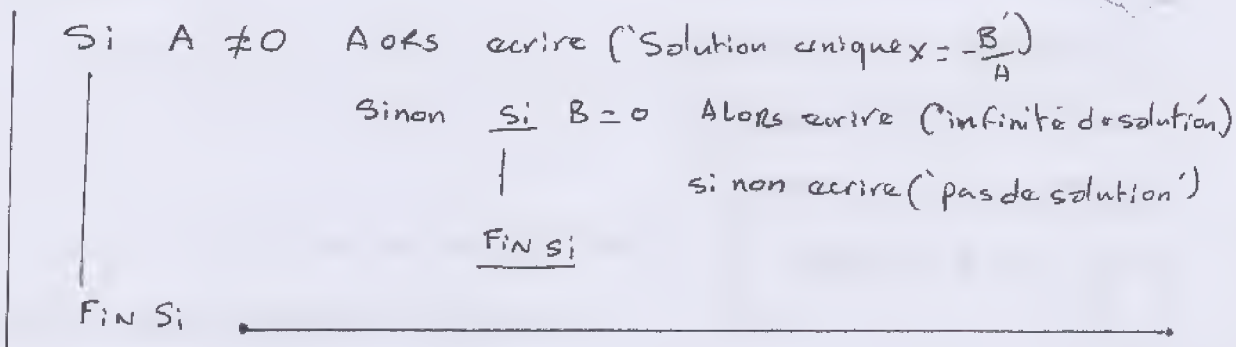
IV -

II Algorithme (Equation 1^{er} degré)

Var : A, B : entier;

x : réel;

Debut



Fin

25/12/2011

* L'ordre des instructions de 3^{ème} partie est très important. Il faut toujours penser à des procédures ouvertes.

II Algorithme (Equation 2^{ème} degré)

Var : A, B, C, Δ : entier; x_1, x_2 : réel;

Debut

Lire (A, B, C);

Si $A = 0$ Alors Repasse Equation 1^{er} degré

sinon $\Delta = B^2 - 4 * A * C$

Si $\Delta < 0$ Alors écrire ('pas de solution')

sinon si $\Delta = 0$ Alors écrire (solution Double)

$$x_1 = x_2 = -\frac{B}{2A}$$

sinon

$$x_1 = \frac{-B - \sqrt{\Delta}}{2A}$$

$$x_2 = \frac{-B + \sqrt{\Delta}}{2A}$$

écrire(x_1, x_2)

Fin Si

Fin Si

Fin Si

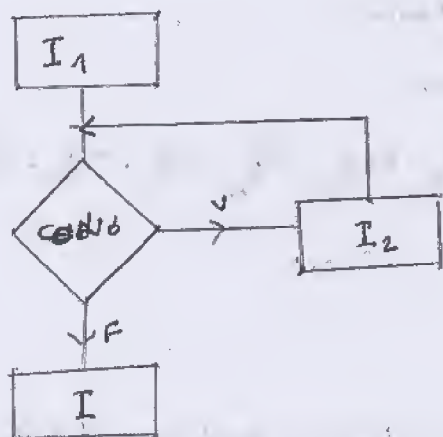
Fin

Les Instruction Itératif:

19/01/2011

Elle nous permet d'exécuter plusieurs fois une ~~inst~~ ensemble d'instruction

La forme général d'une instruction itérative est comme ce schéma:



Exemple : Développer la procédure suivante

$A=2$, $B=0$, $I=1$

Tq $I \leq 4$

Fair $A = 2 * A$

$B = B + 1$

$I = I + 1$

Fait

ecrire (A, B, I)

\Rightarrow

A	B	I
2	0	1
4	1	2
8	2	3
16	5	4
32	26	5

Exercices:

- Donner une procédure qui permet de dilibiré pour un groupe de 30 étudiant qui prépare 5 module diffirante de coéfission 1, 2, 3, 2, 1.
- D'affiché pour chaque étudiant sont matricule, son nom son prénom ses notes, sa moyenne générale et sont résultat (admis ou ajourné)
- D'affiché le nombre des admies la taux de réussite, la moyenne de la classe, la moyenne de chaque module
- Donnez une procédure qui permet de décodé un mot de passe

Solution:

Algorithme délibair

Var

Nom, prénom, Résulta: chaîne de caractère

MAT, Nbre d'étudiant Admi (NBA), I: entier

MG, $N_1, N_2, N_3, N_4, N_5, TR, MG_1, MG_2, MG_3, MG_4, MG_5, MG_6$: Réel

Debut

$NBA = 0; SN_1 = 0; SN_2 = 0; SN_3 = 0; SN_4 = 0; SN_5 = 0; i = 1;$

Tant que $i \leq 30$

Faire Lire (MAT, Nom, Prénom, N_1, N_2, N_3, N_4, N_5);

$SN_i = SN_i + N_i * 5$

$MG = (N_1 * 1 + N_2 * 2 + N_3 * 3 + N_4 * 2 + N_5 * 1) / 9$

Si $MG \geq 10$ Alors Résulta = Admi; $NbA = NBA + 1$;

sinon Résulta = Ajourné;

FIN Si;

crire (MAT, nom, Prénom, N_1, N_2, N_3, N_4, N_5 , Résulta)

$i = i + 1$

Fait;

$MG_1 = SN_1 / 30$

$MG_2 = SN_2 / 30$

$MG_3 = SN_3 / 30$

$MG_4 = SN_4 / 30$

$MG_5 = SN_5 / 30$

$MG_6 = (MG_1 * 1 + MG_2 * 2 + MG_3 * 3 + MG_4 * 2 + MG_5 * 1) / 9$

$TR = (NBA / 30) * 100$

crire (TR, $MG_6, MG_1, MG_2, MG_3, MG_4, MG_5$)

Exercices

- I. Donner une procédure qui permet de vérifier si le nombre A et divise le Nombre B
- II. Donner une procédure qui permet de calculer une division et le rest de la division $\frac{B}{A}$ sont utilisés le mod ni le Div.
- III. Donner une procédure qui permet de vérifier un nombre et premier ou non
- IV. Donner une procédure qui permet de lire son nombre et

① Calculer le Nbre d'éléments positif, négatif et nulle;

② Recherche la valeur "100"

③ Recherche la valeur "100" ou bien "20"

④ Le Nbre d'occurrence de "100"

⑤ La valeur "100" suivie par la valeur "20"

⑥ La valeur "100" et la valeur "20"

⑦ La valeur "100" suivie directement de "20"

solutions

22/01/2019

* Algorithme A Div B ①

Var

A, B, S: entier;

Debut

Lire(A, B);

S := A

Tant que S < B

Faire S := S + A

Fait;

Si S ≥ B alors Ecrire("A divise B")

Fin si; sinon Ecrire("A ne divise pas B")

FIN.

- Cette procédure ne traite qu'un seul cas
celle B > A et A > 0

- Les autres cas qui ne sont pas traités sont:

① A = 0, ② B = 0

③ A et B sont des réels

④ A et B sont les deux négatifs

III Algorithme premier

Debut

Lire (NB)

 $I = 2; TR = V \quad / V = \text{Vrai}, F = \text{Faux} \neq$ Tant que $I < \sqrt{NB}$ Faire si $N \equiv 0 [I]$ Alors $TR = F, I = NB;$

Finsi

 $I = I + 1$

Fait

IV-1

05/02/2011

Debut

 $I = 1, NP = 0, NN = 0, NG = 0$ Tq $I \leq 50$

Faire lire (NB)

Si $NB < 0$ Alors $NG = NG + 1$ Sinon si $NB = 0$ Alors $NN = NN + 1$ Sinon $NP = NP + 1$

Finsi

Finsi

 $I = I + 1$

Fait

 $NP = 50 - (NG + NN)$

Pour une meilleure solution on doit éliminer
 $NP = NP + 1$ et le calculer à la fin

Diffinition

Les procédures peuvent être partagées en 3 catégories

① Le prédécad

C'est les procédures qui ont comme sortie une variable Booléenne (vrais/faux)

② Les fonctions

C'est les procédures qui ont comme sortie un seul résultat

③ Les autres procédures

C'est les autres procédures qui ne sont ni fonction ni prédécad

② Debut

$I = 1$

Tq $I \leq 50$

Faire lire (NB)

Si $NB = 100$ Alors $TR = V, I = 50$

Fin Si

$I = I + 1$

Fait

④ Debut

$I = 1, CP = 0$

Tq $I \leq 50$

Faire lire (NB)

Si $NB = 100$ Alors $CP = CP + 1$

Fin Si

$I = I + 1$

Fait

③ Debut

 $I = 1, TR = F$ TQ $I \leq 50$

Fair Like (NB)

Si NB = 100 ou bien NB = 20 Alors $TR = V, I = 50$

Fini Si

 $I = I + 1$

Fait

⑤ Debut

 $I = 1, TR = F$ TQ $I \leq 50$

Fair Like (NB)

Si NB = 100 Alors $I = I + 1$ TQ $I \leq 50$

Fair Like (NB)

Si NB = 20 Alors $TR = V, I = 50$

Fin Si

 $I = I + 1$

Fait

Fin Si

 $I = I + 1$

Fait

⑥ Debut

$I = 1, TR = F$

TQ $I \leq 50$

Fair Lire (NB).

Si $NB = 100$ Alors $I = I + 1$

TQ $I \leq 50$

Fair Lire (NB)

Si $NB = 20$ Alors $TR = V, I = 50$

Fins:

$I = I + 1$

Fait

Sinon $NB = 20$ Alors $I = I + 1$

TQ $I \leq 50$

Fair Lire (NB)

Si $NB = 100$ Alors $TR = V, I = 50$

Fins

$I = I + 1$

Fait

Fins:

Fais

Exercice:

Ecrire la procédure qui permet de calculer le max, le min et le max et min

Exercice

11/02/2011

Donner une procédure qui permet de lire un nombre entre 1 et 20, si le nombre donné est supérieur strictement à 20 afficher un message plus grand, si inférieur à 1 afficher message plus petit

Solution

TQ TR = F

Faire

Lire (NB)

si $NB > 20$ alors écrire (Le NB est supérieur)sinon si $NB < 1$ alors écrire (Le NB est inférieur)~~non~~ Sinon TR = V

FIN si

FIN si

Fait

Exercice

Trouver une procédure qui permet de décoder un code au max après 3 tentative

Solution

Lire (code)

Lire (chaines)

 $i = 1$ TQ ($i < 4$) et ($chaines \neq code$)

faire lire (chaines)

 $i = i + 1$

fait

si $i \geq 4$ Alors Bloquer

exercice

Donner une procédure qui permet de lire un Nbre quel que soit et
s'afficher sa table de multiplication jusqu'à 10.

Beut

Lire (NB)

$I = 1$

Tq $I \leq 10$

Faire écrire ($NB * I = NB * I$)

$I = I + 1$

Fait

19/02/2011)

Les Vecteurs: est une structure de donnée qui ^{nous} permet de stocker des informations **Contigue**, chaque valeur est désignée par les paramètres suivants

Contigue: متجاورين

① Le nom de vecteur

② La dimension de Vecteur N (Le Nbre de case réservé)③ Le Nbre de case Remplie M Exemple:

T =	3	8	4	0	10	
	1	2	3	4	5	6

* Le Nom de ce vecteur c: T

* La dimension de ce vecteur c'est: 6

* Le Nbre de case Remplie c'est: 5

* $T[3] = 4$ * $T[5] = 10$ * $T[6] = "$ " * $T[7] = \text{erreur}$ * $T[T[1]] = 4$ Remarque* C'est le $m = 0$ le vecteur est vide* C'est le $m = N$ le vecteur est plein

* C'est le vecteur est vide on ne peut faire ni la recherche ni la suppression

* C'est le vecteur est pleine on ne peut pas insérer

* Un vecteur n'est pas redondant si seulement si aucune des informations ne répète + qu'il fail

④ Les Trois procédures de base pour n'importe quelle base de données sont:

- La recherche d'une valeur Val.

- Supprimer une valeur Val.

- Insérer une valeur Val

Détaillant ces trois procéduresDans le cas : Le vecteur n'est pas trié

① Recherche "O" [T, N, M, Val, TR, K]

• Debut

TR = F

I = 1

Tq $I \leq M$ Faire si $T[I] = \text{Val}$ Alors TR = V, K = I, E = M~~Fais~~ Finsi

I = I + 1

Fait.

② Suppres "O" [T, N, M, Val, TR, K]

• Debut (Recherche "O") Refaire la Procédure de la Recherche

si TR = V Alors $T[K] = T[M]$, $M = M - 1$

Finsi

③ Insertion "O" [T, N, M, Val, TR, K]

Pour insérer une valeur "Val" il faut vérifier les 2 conditions suivantes

① La valeur Val n'existe pas

② La base de données n'est pas pleine (Le vecteur n'est pas plein)

 $(M < N)$

• Debut

si $M < N$ Alors Rech "O" [T, N, M, Val, TR, K]si TR = Faux Alors $M = M + 1$, $T[M] = \text{Val}$

Finsi

Finsi

Fin

cette procédure nous permet d'insérer une valeur "Val" dans un vecteur V

Exercices

26/02/2011

- ① Donner la procédure qui permet de trouver le PGCD de deux nombres
- ② Un client a déposé un million de dinars dans une banque avec un taux d'intérêt de 10% (Annuel)
- donner une procédure qui permet de trouver le solde de compte à pour les 25 premières années.
- ③ Un client a demandé un crédit trois millions de dinars avec un taux d'intérêt de 12% (Annuel) la capacité de remboursement de ce client est (270 000,00 DA par année)
- ⓑ 360 000,00 DA par année
 - ⓒ 480 000,00 DA par année
- * étudier la rentabilité dans ces cas
 - * Trouver le temps qu'il faut pour rembourser son crédit
- ④ même exercice donnée que la question ③
- * La capacité de remboursement c'est 40 000,00 DA par mois
- ⑤ La capacité de remboursement d'un client est de 60 000,00 DA par mois sur une durée de 30 ans avec le taux d'intérêt de 12%
- * Trouver le capital qu'on doit lui accorder
- ⑥ Un client a demandé un crédit de 3 000 000,00 de DA sur une durée de 25 le taux d'intérêt est de 12%
- * Trouver la valeur qu'il doit rembourser soit (Annuel ou par mois)

CACI-2011/2012 Informatique (USTHB)Module SDDEXERCICE 1:

Donner une procédure qui permet de trouver la durée entre deux dates données (jj1, mm1, aa1) et (jj2, mm2, aa2)

EXERCICE 2:

A/ Ecrire un algorithme qui demande un nombre de départ, et qui ensuite écrit la table de multiplication de ce nombre, présentée comme suit (cas où l'utilisateur entre le nombre 7) :

Table de 7 :

$$7 \times 1 = 7$$

$$7 \times 2 = 14$$

$$7 \times 3 = 21$$

...

$$7 \times 10 = 70$$

B/ Ecrire un algorithme qui demande un nombre de départ, et qui calcule la somme des entiers jusqu'à ce nombre. Par exemple, si l'on entre 5, le programme doit calculer :

$$1 + 2 + 3 + 4 + 5 = 15$$

NB : on souhaite afficher uniquement le résultat, pas la décomposition du calcul.

EXERCICE 3:

Donner une procédure qui permet de chercher s'il existe les deux nombres N1 et N2 dans un vecteur ordonné. (La plus optimum possible)

Bonne chance!

M.A. BOUZEGHOUB

Solution du sujet (USTHB)

Le 12/03/2011

Exercice 1

calculer la durée des deux dates différentes: $Durée = D_2 - D_1$

* $D_1 = JJ_1 / MM_1 / AA_1$ * $D_2 = JJ_2 / MM_2 / AA_2$ * Durée: AA, MM, JJ

Cette procédure se compose de 2 solutions

① Début

si $JJ_1 > JJ_2$ Alors $JJ_2 = JJ_2 + 30$, $MM_2 = MM_2 - 1$
 FinSi

si $MM_1 > MM_2$ Alors $MM_2 = MM_2 + 12$, $AA_2 = AA_2 - 1$
 FinSi

$AA = AA_2 - AA_1$

$MM = MM_2 - MM_1$

$JJ = JJ_2 - JJ_1$

② Début

$AA = AA_2 - AA_1$

$MM = MM_2 - MM_1$

$JJ = JJ_2 - JJ_1$

si $JJ < 0$ Alors $JJ = JJ + 30$, $MM = MM - 1$
 FinSi

si $MM < 0$ Alors $MM = MM + 12$, $AA = AA - 1$
 FinSi

Exercice 2:

① Lire(NB)

$I = 1$ Tant que $I \leq 10$

Faire écrire (NB * I = NB * I)

fait $I = I + 1$

② Début

Lire(NB)

$I = 1$, $S = 0$

Tant que $I \leq NB$

Faire $S = S + 1$, $I = I + 1$

Fait

① Il ne faut pas tester pour affirmer de la valeur n'existe pas

↗ existe
↘ existe
② $\frac{1+M}{2}$

② Le Nbre de teste qu'il nous faut pour recherché une valeur Val dans le cas on à 50% de chance de la trouvé égal $\frac{M + \frac{1+M}{2}}{2}$

exemple: si on a $M = 1000000 \Rightarrow \frac{1000000 + \frac{1+1000000}{2}}{2} = 750000$

* Un Vecteur Trier

C'est un Vecteur Trier si seulement si il verifie les condition suivante

$$\forall I < M \quad T[I] < T[I+1]$$

* Les Trois pocéure de base dans un Vecteur Trier

① Rech^{re} [T, N, M, Val, TR, K]

debut TR = F

```

    si T[i] ≤ Val et Val ≤ T[M]
    Alors I = 1 Tq T[i] < Val
        Fair I = I + 1
        fait
    si T[i] = Val
        Alors TR = V, K = I
    finsi
  
```

FIN

Remarque

Après chaque teste on supprime un seul élément

La Recherche DIKHOTOMIQUE:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
2	4	6	8	10	11	15	18	20	23	30	36	40	45	50	60	70

$Mil = \frac{inf + sup}{2}$ Mil = milieu * inf = inférieur * sup = supérieur

① $T[Mil] = Val$ ② $T[Mil] > Val$, $sup = mil - 1$ ③ $T[Mil] < Val$, $inf = Mil + 1$

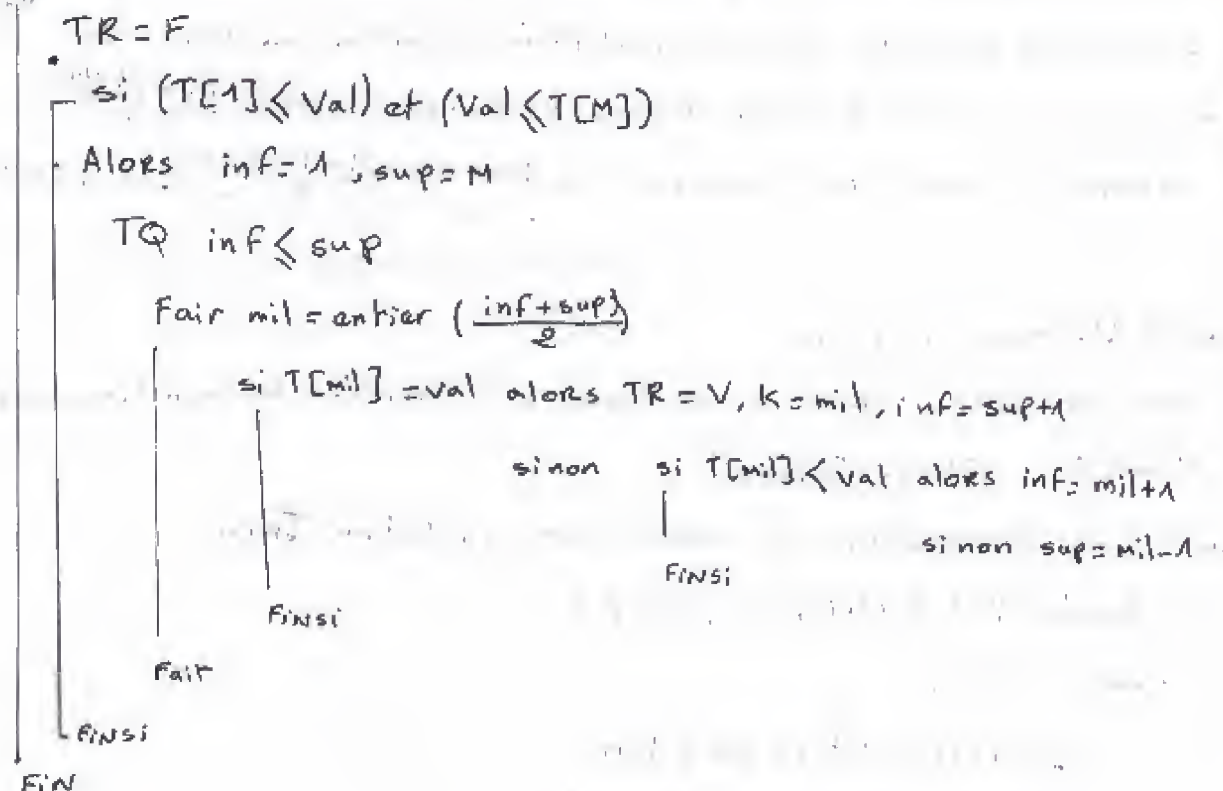
Fait la recherche de : 0, 80, 40, 8, 2, 70, 33

Recherche dichotomique

02/04/2011

Rech dich [T.N.M.Val.TR.K]

Debut

Etude Théorique de la procédure Recherche dichotomique

- * On peut remplacer l'instruction $mil \leftarrow mil = \frac{inf + sup}{2}$ par $mil = \frac{mil + sup}{3}$ ou par $mil = \frac{inf + sup}{2}$
- * Les des instruction $inf = mil + 1$ et $sup = mil - 1$ ne peuvent pas être remplacé par $inf = mil$ et $sup = mil$
- * La condition $inf \leq sup$ ne peut pas être remplacé par $inf < sup$ par ce que "Les cas ou il reste une seul case"

Insertion une valeur Val dans un vecteur trier:

Pour inséré une valeur dans un vecteur trier il faut vérifier les condition suivante:

- ① $M < N$ (Le vecteur n'est pas plein)
- ② La valeur Val n'existe pas.

* L'orde de ces condition est important

③ Il faut insérer la valeur val à sa place pour garder le vecteur trier

Insertion [T, N, M, Val]

Debut

si $M < N$ Alors Recherche Dich [T, N, M, Val, TR, K]

si $TR = F$ Alors $T[M] > Val$

Fair $T[M+1] = T[M]$

$M = M - 1$

Fait

09/04/2011

Correction des devoirs:

Exercice N°1:

Donner une procédure qui permet de calculer le signe du produit des éléments d'un vecteur

1^{ère} solution

$P = 1, I = 1$

TQ $I \leq N$ Fair

$P = P * T[I], I = I + 1$ Fait

si $P < 0$ Alors produit négatif.

sinon si $P > 0$ alors produit positif

sinon produit nul

2^{ème} solution

$S = 1, I = 1$

TQ $I \leq N$

Fair si $T[i] = 0$ Alors $sig = 0$ alors

sinon si $T[i] < 0$ Alors $S = S * (-1)$

Fin si

Fin si

écrire Signe (S)

Exercice N° 2

Donner une procédure qui permet de vérifier la validité d'un chèque (la date d'un chèque) * Le chèque est valable une année

solution 1

* D_1 (Si égalité du chèque) JJ MM AA

D_2 (jour)

JJ MM AA

Durée (D_2, D_1) AA MM JJ

solution 2

Si $AA > 0$ Alors le chèque n'est pas valide sinon le chèque est valide

Si $(AA_1 = AA_2)$ ou $(AA_2 = AA_1 + 1 \text{ et } MM_2 < MM_1)$ ou $(AA_2 = AA_1 + 1 \text{ et } MM_2 = MM_1)$

et $JJ_2 < JJ_1$ Alors le chèque est valide sinon le chèque n'est pas valide

Exercice N° 3

Donner une procédure qui permet de chercher la valeur 3 suivie de la valeur 5 dans un vecteur

exercice N° 4

Un client à déposer 1 million de DA dans une banque, avec un taux d'intérêt 7%

Un autre client à déposer 0,8 Million DA dans autre banque, avec intérêt de 10%

Donner une procédure qui permet de déterminer la durée pour que les deux capitaux doit être égaux

$C_1: 1\,000\,000 \text{ DA} \rightarrow 7\%$

$C_2: 800\,000 \text{ DA} \rightarrow 10\%$

$C_1 = 10^6$

$I = 0$

Tq $I \leq 20$

Faire $C_1 = C_1 \cdot 1.07$

$I = I + 1$

Fait

$I = 0$

$C_1 = 10^6$

$C_2 = 8 \cdot 10^5$

Tq $C_2 < C_1$

Faire $C_1 = C_1 \cdot 1.07$

$C_2 = C_2 \cdot 1.1$

$I = I + 1$

Fait

Exercice I

16/01/2011

Donnez une procédure qui permet de rechercher les deux valeurs (3 et 5) (3 ou bien 5) dans un vecteur d'une dimension N (N case remplie) dans les cas suivants:

- ① Le vecteur T n'est pas ordonné
- ② Le vecteur T est ordonné

Exercice II

Donnez une procédure qui permet de supprimer la plus grande valeur et la plus petite valeur dans un vecteur T , dans les cas suivants:

- ① Le vecteur T n'est pas trié
- ② Le vecteur T est trié

Solutions:Exercice I

- ① Cherche la valeur 3 ou bien 5 dans un vecteur T non ordonné

La recherche est une recherche ~~serielle~~ séquentielle par ce que le vecteur n'est pas trié

Début

 $i = 1, TR = F$ TQ $i \leq M$ Fais si $(T[i] = 3 \text{ ou bien } T[i] = 5)$ Alors $TR = V, i = M$

Fin si

 $i = i + 1$ fais
Fin

- ② Chercher la valeur 3 et 5 dans un vecteur non ordonné

Début $i = 1; TR = F$ TQ $i < M$

Faire si $T[i] = 3$ Alors $i = i + 1$

TQ $1 \leq i \leq M$

Faire si $T[i] = 5$ Alors $TR = V, i = M$

FIN si
 $i = i + 1$

fait
sinon si $T[i] = 5$ Alors $i = i + 1$

TQ $1 \leq i \leq M$

Faire si $T[i] = 3$ Alors $TR = V, i = M$

FIN si
 $i = i + 1$

fait

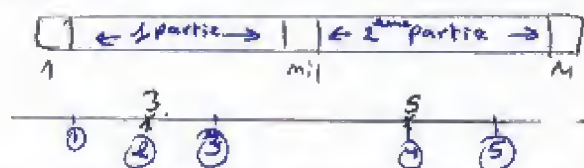
FIN si

FIN si

④ La recherche de la valeur 3 ou bien 5 dans un vecteur trier
4 cas

- ①. si $T[1] > 5$ ou bien $T[M] < 3$ il n'y a ni 3 ni 5
- ②. si $T[1] > 3$ et $T[M] > 5$ Alors on cherche "5" par la méthode dichotomique
- ③. si $T[1] < 3$ et $T[M] < 5$ Alors on cherche "3" par la méthode dichotomique
- ④. si $T[1] < 3$ et $T[M] > 5$ Alors chercher les deux valeurs "3" ou "5"

* Détaillant le cas N° 4 qui nous permet de chercher la valeur N° 3 ou bien "5"



• Si on est dans le cas N° ④ on va chercher le 3 ou 5 dans la 2^{ème} partie du vecteur

• Si on est dans le cas N° ② Le Nombre est exacte et on arrête

• Si on est dans le cas N° ③ on va chercher le 3 dans la 1^{ère} partie, si il existe on arrête si non on cherche le 5 dans la 2^{ème} partie

• Si on est dans le cas N° ④ le Nombre est exacte et on arrête

• Si on est dans le cas N° ⑤ on va chercher le 3 ou 5 dans la 1^{ère} partie

inf = 1, sup = m

TQ $\text{inf} \leq \text{sup}$

Faire $\text{mil} = \text{entier} \left(\frac{\text{inf} + \text{sup}}{2} \right)$

si $T[\text{mil}] < 3$ alors $\text{inf} = \text{mil} + 1$

sinon si $T[\text{mil}] = 3$ Alors $\text{TR} = V$; $\text{inf} = \text{sup} + 1$

sinon si $T[\text{mil}] < 5$ Alors \Rightarrow

suivre Recherche dikho $[T, N, M, \text{inf}, \text{mil} - 1, 3, \text{TR}, K]$

si $\text{TR} = V$ alors $\text{inf} = \text{sup} + 1$

sinon Recherche dikho $[T, N, M, \text{mil} + 1, \text{sup}, 5, \text{TR}, K]$

sinon si $T[\text{mil}] = 5$ Alors $\text{TR} = V, \text{inf} = \text{sup} + 1$

sinon $\text{sup} = \text{mil} - 1$

Fins;

Fins;

FINS;

Fait

D) Recherche de la valeur 3 et 5 dans un vecteur TR

inf = 1, sup = m

TQ $\text{inf} \leq \text{sup}$

Faire $\text{mil} = \text{entier} \left(\frac{\text{inf} + \text{sup}}{2} \right)$

si $T[\text{mil}] < 3$ alors $\text{inf} = \text{mil} + 1$

sinon si $T[\text{mil}] = 3$ Alors Rech deko $[T, N, M, \text{mil} + 1, \text{sup}, 5, \text{TR}, K]$

sinon si $T[\text{mil}] < 5$ Alors \Rightarrow

\Rightarrow Rech dikho $[T, N, M, \text{inf}, \text{mil} - 1, 3, \text{TR}, K]$

si $\text{TR} = V$ Alors Rech dikho $[T, N, M, \text{mil} + 1, \text{sup}, 5, \text{TR}, K]$

sinon $\text{inf} = \text{sup} + 1$

sinon si $T[\text{mil}] = 5$ Alors Rech dikho $[T, N, M, \text{inf}, \text{mil} - 1, 3, \text{TR}, K]$

sinon $\text{inf} = \text{sup} + 1$

Fins;

Exercice N° III

Donnez une procédure qui permet de rechercher les trois valeurs suivant 3, 5, 10 dans un vecteur Trier

① déterminer les différents cas

② $T[1] > 10$ ou bien $T[M] < 3$ il n'existe aucun nombre

③ Le dernier cas

$T[1] < 3$ et $T[M] > 10$

* Solution exercice N° II (page 37)

① Le vecteur T n'est pas Trier

Rech max [] \rightarrow max = k

Rech min [] \rightarrow min = L

$T[k] = T[M]$

$T[L] = T[M-1]$

$M = M - 2$

② Le vecteur T n'est pas Trier

$T[1] = \text{min}$

$T[M] = \text{max}$

$I = 1$

TQ $I < (M-1)$

Faire $T[I] = T[I+1]$

$I = I + 1$

Fait

$M = M - 2$

* Ces deux procédures si elles sont correctes si le vecteur n'est pas redondant

* Dans le cas ou le vecteur est redondant, on peut avoir plusieurs min et plusieurs max. Alors on peut supprimer jusqu'à tous le vecteur

1	1	2	2	2
---	---	---	---	---

Exercice:

Donnez une procédure qui permet de supprimer de valeur val_1, val_2 dans les deux cas suivant :

- ① Trier
- ② Non Trier
- ③ Redondant
- ④ Non Redondant

23/04/2011

Exercice : On a 5 chapeaux

 $E = \{R_1, R_2, R_3, B_1, B_2\}$

On place 3 personnes de face

. Comment

1 Suppression d'un élément dans un vecteur Trier

Pour supprimer un élément dans un vecteur Trier, il faut que l'élément existe dans ce vecteur

Suppr [T, N, M, Val]

(Gtiq 1)
debut Rech Diko [T, N, M, Val, TR, K]

si TR = V Alors I = k

TQ I < M

Faire T[I] = T[I+1]

I = I + 1

Fait

M = M - 1

(Aller à Gtiq 1) → pour le vecteur Redondant

Fin si

FIN

Remarque:

Cette procédure est correcte que si est non redondant

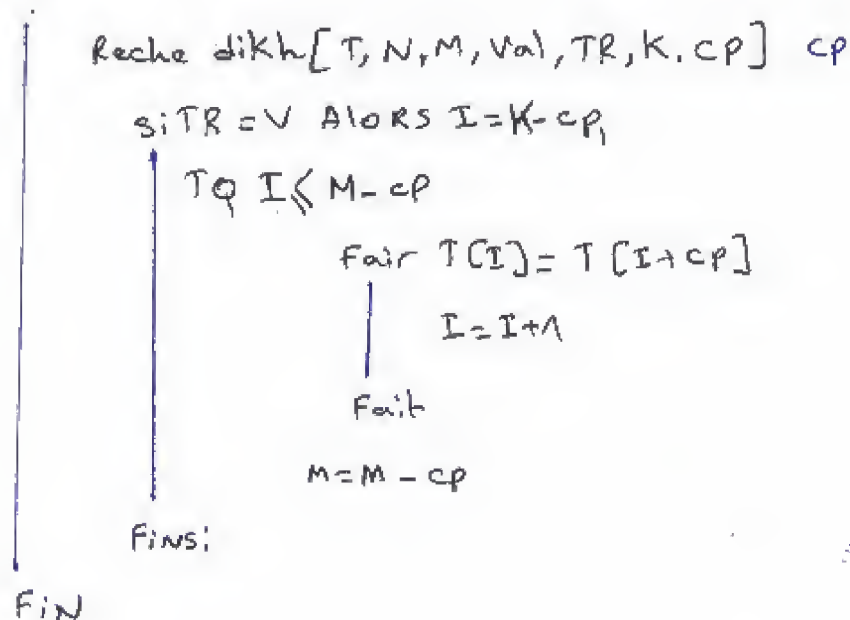
Exercice:

Donnez un procédure qui permet de supprimer une valeur val dans un vecteur trier et redondant

* C'est la solution la m du la suppression on ajoutant
Les (Aller à Gtiq 1)

- On modifier la procédure recherche dichotomique de telle sorte qu'elle nous donne $\text{RechDikh}[T, N, M, \text{Val}, \text{TR}, K, \text{CP}]$
- avec la recherche dichotomique si $\text{TR} = \text{V}$ alors $T[K] = \text{Val}$, les autres valeurs se trouvent avant ou après la valeur $\text{Val}(T[K])$
- CP c'est le nombre d'occurrence de la valeur val
- * $\text{Sup2}[T, N, M, \text{Val}]$

debut



Les Triés

Le travail dans une base de données triée est plus efficace sur une base de données non triée ce qui justifie le tri des bases de données

Il existe plusieurs méthodes de trier une base de données

méthode 1: le tri par maximum

1^{ère} étape: $\text{sup} = M$ (sup = supérieur)

2^{ème} étape: $\text{Max}[T, 1, \text{sup}, K]$

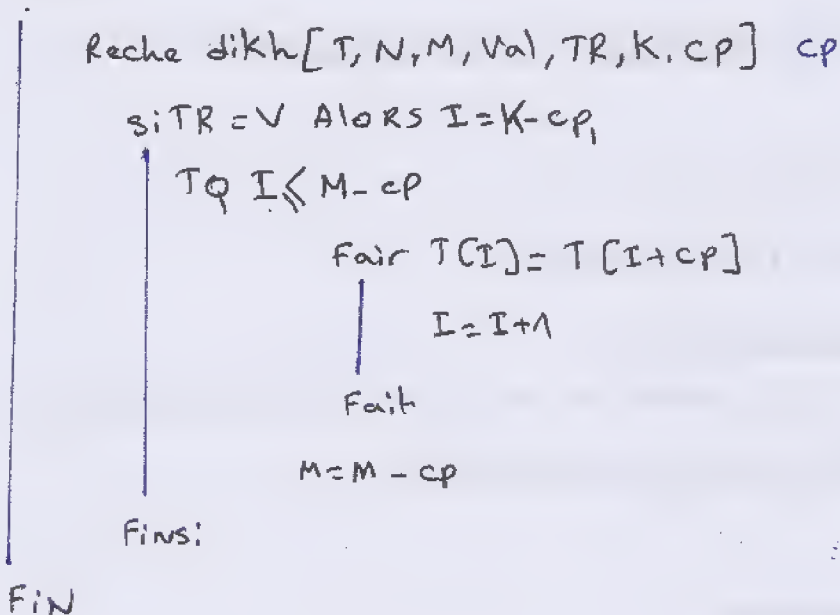
$T[K] \rightsquigarrow T[\text{sup}]$

3^{ème} étape: $\text{sup} = \text{sup} - 1$

* Aller à la 2^{ème} étape

- On modifier la procédure Recherche dichotomique de telle sorte qu'elle va nous donner $\text{Rechdikh}[T, N, M, \text{Val}, \text{TR}, K, \text{CP}]$
 - avec la recherche dichotomique si $\text{TR} = \text{V}$ alors $\text{T}[K] = \text{Val}$, les autres valeurs se trouvent avant ou après la valeur $\text{Val}(\text{T}[K])$
 - CP c'est le nombre d'occurrence de la valeur val
- * $\text{Sup2}[T, N, M, \text{Val}]$

debut



Les Triés

Le travail dans une base de données Triée est plus efficace sur une base de données non Triée ce qui justifie le Trié des bases de données

Il existe plusieurs méthodes de Trier une base de données

méthode 1: Le trié par maximum

1^{ère} étape: $\text{sup} = \text{M}$ ($\text{sup} = \text{supérieur}$)

2^{ème} étape: $\text{Max}[T, 1, \text{sup}, K]$

$\text{T}[K] \rightsquigarrow \text{T}[\text{sup}]$

3^{ème} étape: $\text{sup} = \text{sup} - 1$

* Aller à la 2^{ème} étape

Fair si $T[I] > T[I+1]$ Alors Permut $[T[I], T[I+1]]$; $TR = V$

Finsi

$I = I + 1$

Fait

$m = m + 1$

si $TR = V$ Alors Aller à Etiqu

Exercice:

Donner une procédure qui permet de supprimer tous les éléments redondants dans une base de données (un vecteur)

Solution

I Trier d'abord le vecteur

II ① si $T[I] \neq T[I+1]$ Alors $T[I]$ n'est pas redondant

② si $T[I] = T[I+1]$ Alors $T[I]$ est redondant

III déterminer le nombre de redondance $T[I]$ (Les $T[I]$ sont en bloque) = K

IV écrasé (décaler le vecteur à gauche de $(K-1)$ case)